Sensors and Materials, Vol. 37, No. 5 (2025) 2061–2080
MYU Tokyo

S & M 4042

Automated AI Approach for Noninvasive Shrimp Length and Weight Estimation Using Underwater Imaging and Feeding Induction

Shao-Yong Lu,* Yu-Sheng Tu, and Wen-Ping Chen

Department of Electrical Engineering National Kaohsiung University of Science and Technology Kaohsiung, No. 415, Jiangong Rd., Sanmin Dist., Kaohsiung City 807618, Taiwan.

(Received January 27, 2025; accepted April 25, 2025)

Keywords: image processing, object detection, convolutional neural network, shrimp body length

We aim to develop an objective method for measuring shrimp length in traditional white shrimp farming using images from an automated underwater camera system, reducing reliance on subjective assessment. Traditional manual methods often suffer from subjectivity and inaccuracies, leading to inefficient feed management strategies. Computer vision techniques were used for object detection and image preprocessing. A deep learning network was used to classify completeness and assess the length and weight of shrimps. We also analyzed the correlation between the body length and weight of shrimps. The dataset consists of 8401 images categorized as measurable (3112) and visible (5289). An accuracy of 95.0% was obtained with an average error rate of 8.4%, highlighting the effectiveness of the proposed method. The weight estimation exhibited an average error rate of 22%. This system can optimize feed management and enhance aquaculture sustainability, reducing both resource waste and operational costs. Furthermore, we utilized underwater cameras as sensing devices, combined with specially designed feeding platforms and observation materials, to enable real-time image-based monitoring and noninvasive shrimp measurement.

1. Introduction

Fish are essential food for human growth, development, and health owing to their nutrient content. Fisheries and aquaculture are also vital for economies and livelihoods globally. The State of World Fisheries and Aquaculture of the Food and Agriculture Organization of the United Nations reported that aquaculture production now exceeds capture fisheries. Global fisheries face challenges from competition and climate change, making AI solutions increasingly valuable as the industry also struggles with an aging workforce and rising costs.[1] Aquaculture, particularly shrimp farming, plays a crucial role in global food security. However, traditional farming methods face significant challenges, including inaccurate length measurement, which impacts feed management and growth predictions. AI-based automated systems provide a promising solution to these issues.

*Corresponding author: e-mail: i112154102@nkust.edu.tw
https://doi.org/10.18494/SAM5564

ISSN 0914-4935
https://myukk.org/

This study is motivated by the rising significance of intelligent systems in aquaculture, as shown by these developments. For effective management, it is crucial to accurately measure shrimp size and weight. Measurement inaccuracies may result in the adoption of suboptimal aquaculture practices, causing financial setbacks. Since the growth rate of white shrimp is affected by factors such as stocking density, species, environment, and feeding amounts, precise and timely monitoring is essential for optimizing production outcomes. In our system, underwater imaging cameras function as primary sensors to capture environmental and biological data. Specialized observation platforms and feeding devices serve as supporting materials, optimizing the sensing environment and enabling accurate data acquisition for automated analysis.

Traditional observation methods involve the real-time monitoring of the feeding tray, as shown Fig. 1, which consists of simple mechanical structures and materials. By observing these trays, farmers can guess shrimp health, environmental adaptation, and feed adequacy, and make the necessary management adjustments.[2] Manual feeding tray observations are labor-intensive and prone to errors. Inaccurate shrimp length estimation can lead to overfeeding or underfeeding, affecting aquaculture productivity and increasing operational costs.

AI and IoT have been successfully applied in aquaculture, such as automated fish health monitoring, real-time feed optimization, and environmental condition tracking, demonstrating their potential for shrimp farming. With the aid of these technologies, aquaculture monitoring systems, employing cameras to capture images,[3] enable the comprehensive observation and assessment of diverse water quality parameters. Productivity can be increased by controlling automated feeding devices, feed conversion ratios, and health status referring to the results of monitoring.

Image processing techniques, such as segmentation and background subtraction, enable precise shrimp length measurement by addressing underwater challenges such as varying light conditions and turbidity.[4] Real-time analysis of surface activity is achieved by transmitting

Fig. 1. (Color online) Conventional method for measuring shrimp length.

images captured from cameras located above the tanks to a central server for processing and recognition, as illustrated in Fig. 2. However, bottom-dwelling species cannot be observed. Image processing techniques, such as segmentation and feature extraction, are critical for addressing underwater visibility issues in shrimp length measurement.

In this study, we aim to develop an AI-based automated system for shrimp length measurement, addressing limitations in traditional manual methods and contributing to efficient aquaculture management. In this research, we used a new method for measuring the length of white shrimp: an automated system that uses underwater cameras and advanced deep learning models (a visual representation of this process can be found in Fig. 3) to ensure both accuracy and efficiency in our measurements, as detailed further in Ref. 5.

To successfully capture images of shrimp, a feeding platform, as shown in Fig. 4, was specifically designed, and a sophisticated algorithm capable of detecting moving objects was also developed. To pinpoint the shrimp in the images, a shrimp recognition model was trained using a convolutional neural network (CNN),[6] and subsequent image processing techniques were employed to detect shrimp from the background and measure their length.

The images taken by traditional underwater cameras are affected by suspended particles, compromising image clarity, especially in turbid water.[7] To mitigate this, an airtight chamber filled with air was designed to provide an appropriate shooting distance even in murky water, as shown in the lower left image of Fig. 4. An observation board with bait was placed in front of the transparent window to attract aquatic animals, illuminated by waterproof lighting modules from both sides. This setup ensured a clear, top-down image of aquatic activity, with the observation

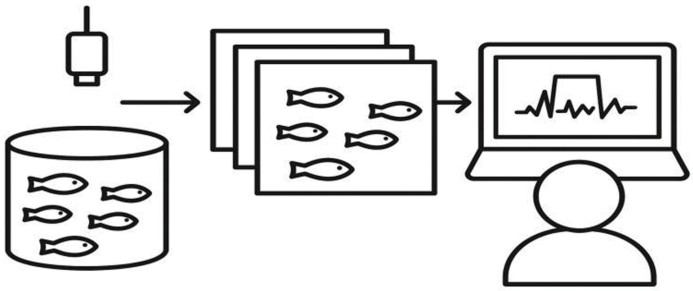


Fig. 2. Real-time monitoring and feeding decisions.

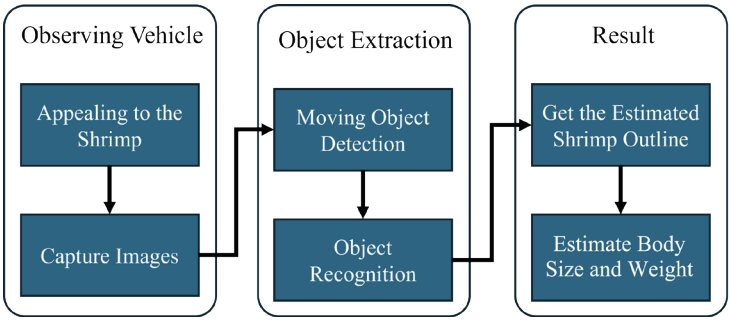


Fig. 3. (Color online) Process of measuring shrimp length.

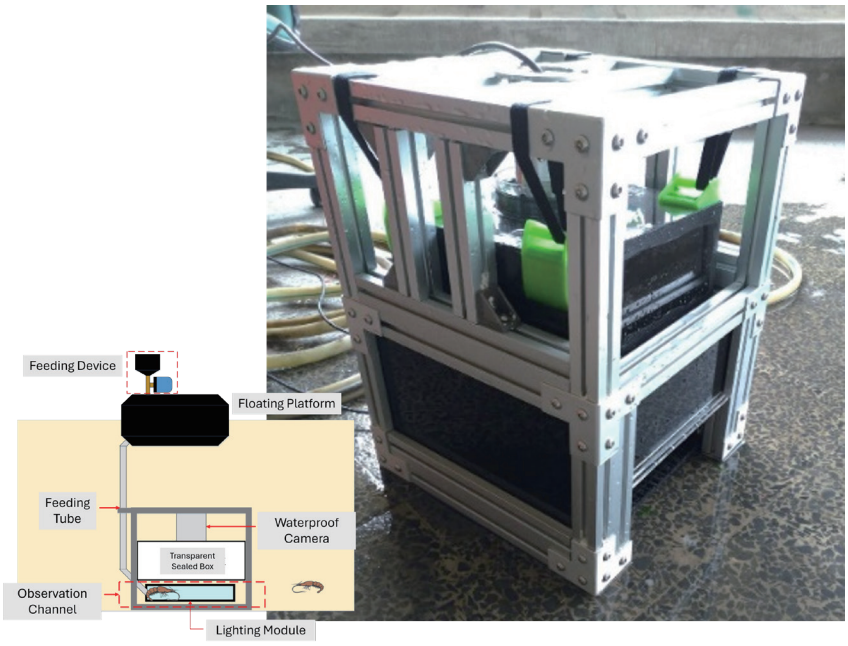


Fig. 4. (Color online) Feeding platform.

channel height limiting shrimp vertical movement to maintain image clarity. Displayed in the right-hand side image of Fig. 4 is the feeding platform, which is clearly visible in the photograph.

A system for feeding was created, as illustrated in Fig. 5, with the purpose of attracting shrimp to the area designated for observation. The device was mounted on a floating platform that consisted of a feed bucket, a normally closed stainless steel electric ball valve, a feed tube connected to the underwater area, and an electromagnetic switch for control.[8] The intermittent control of the electromagnetic switch ensured stable delivery of feed, preventing water quality degradation.[9] As shown in Table 1, the optimal feeding duration is achieved through a process with distinct opening and closing phases, each lasting 5.5 and 6.5 s, respectively.

As illustrated in Fig. 5, a specifically designed and installed feeding mechanism was implemented to lure shrimp into the observation area, thereby facilitating their observation and subsequent study. A floating platform, composed of a feed bucket, a normally closed stainless steel electric ball valve, a feed tube extending to the underwater area, and an electromagnetic switch for control, served as the mounting structure for the device. The consistent delivery of feed, a critical factor in preserving the water purity, was accomplished by implementing an electromagnetic switch, overseen by an intermittent control system; this successfully prevented any contamination from occurring.[10] For optimal results, refer to Table 1, which details the recommended feeding duration, namely, 5.5 s for opening and 6.5 s for closing the feed tube valve.

2. Data, Materials, and Methods

The system integrates average background modeling for underwater object segmentation and CNNs for accurate shrimp detection, overcoming the challenges of water turbidity and dynamic lighting conditions.

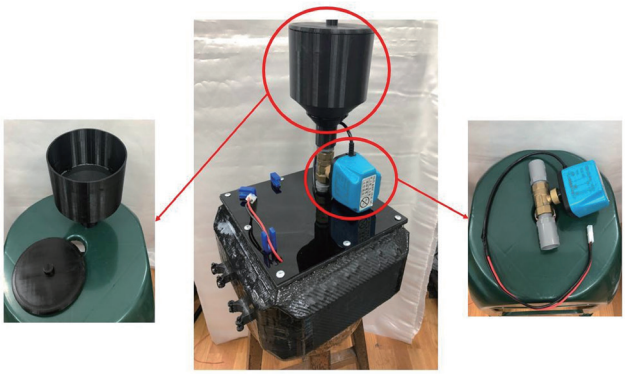


Fig. 5. (Color online) Feeding mechanism.

Table 1
Record of feeding volume experiment.

Open/Close (s)	3.5/4.5	4.5/5.5	5.5/6.5	6.5/7.5
Test 1	3.1	4.5	4.8	5.2
Test 2	3.5	4.4	5.2	5.6
Test 3	2.9	3.9	5.1	5.0
Test 4	3.4	5.0	4.9	5.5
Test 5	3.8	4.9	5.0	6.1
Average amount (g)	3.31	4.38	4.99	5.75
Standard error (g)	0.3	0.4	0.1	0.4

2.1 Preliminary

Moving object detection and tracking algorithms are crucial in computer vision,[11] and related technology is rapidly developing and being used in diverse applications, such as military guidance, visual navigation, video surveillance, and intelligent transportation systems. In moving object detection, optical flow, frame differencing, background subtraction, and CNNs are mainly used. In object detection and tracking algorithms, moving objects are detected and their trajectories are predicted. By using advanced computing, communication, and image processing technologies, moving object detection and tracking are enabled through image processing and pattern recognition and used in military guidance, visual navigation, video surveillance, intelligent transportation, medical diagnostics, and industrial product inspection.[12]

We implemented background subtraction to isolate shrimp from the water background. Although optical flow was tested,[13] it was found unsuitable owing to sensitivity to water turbulence and inconsistent lighting. The process involves segmenting the initial image from continuous frames to determine the optical flow field distribution, followed by gradient-based dynamic vector calculation. This method effectively describes pixel brightness changes due to object movement, enabling the segmentation of moving objects.[14] Background subtraction involves creating a fixed background image and subtracting it from subsequent image sequences to identify moving objects.[15] While effective in stable backgrounds, this method falters with fluctuating light sources or sudden object movements, leading to detection errors.

CNNs use deep learning for enhanced learning. Convolutional layers extract features; fully connected layers classify them in CNNs. The model was designed to segment shrimp contours and measure lengths accurately, addressing issues in manual measurements. Convolution operations are used in image processing to remove noise, sharpen images, and detect edges. The process involves selecting a point in an image, defining an $n \times n$ range around it, assigning weights (convolution kernels), and summing the weighted values for all pixels. This process extracts features and forms a feature map, which is refined during network learning for accurate classification.[16]

Following the convolutional layer, pooling layers reduce data volume and highlight features. In pooling, the maximum value within an $n \times n$ range is selected to retain significant features while reducing data and computational load. After feature extraction, the fully connected layer classifies these features. Each feature is weighted, with significant features receiving higher weights. This mechanism ensures accurate classification based on the extracted feature's prominence. In a multilayer neural network, input data is computed at each layer using a weighting matrix W.

$$y_1 = W_1 x \tag{1}$$

$$y_2 = W_2 y_1 \tag{2}$$

$$y_2 = W_2 y_1 = W_2 W_1 x = (W')x \tag{3}$$

Dropout prevents overfitting by randomly "turning off" nodes during training. Dropout ensures that each training pass activates different nodes and enhances the model's generalization and robustness.

Activation functions introduce nonlinearity into the model to prevent it from collapsing into a single layer of linear operations. The activation functions Sigmoid,[17] Hyperbolic Tangent,[18] and ReLU[19] are frequently utilized, and Softmax[20] serves as another activation function that normalizes the final output layer into a probability distribution that sums to unity, thus allowing for effective classification.[21]

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} . j = 1, \ldots, K \tag{4}$$

2.2 Moving object detection method

The stability and balance of the aquatic environment are crucial for shrimp farming. Water color is an indicator of the algal population that affects the efficacy of moving object detection methods. We combined average background modeling and CNN-based neural networks to address issues related to water color variability.

2.2.1 Average background modeling

Moving object segmentation was achieved using average background modeling, a technique that dynamically updates the background model by incorporating ongoing object analysis results. As illustrated in Fig. 6, the process facilitates the discrimination between foreground and background elements, subsequently updating the background model whenever an object maintains a stationary position for a predetermined duration threshold.[22]

2.2.2 Image preprocessing

The initial dataset, composed of 100000 raw images, underwent a rigorous cleaning process involving the removal of duplicate entries, images with low resolution, and data points deemed nonrepresentative, ultimately yielding a refined dataset of 8401 high-quality images. By using this approach, we can be certain that the model is trained on a diverse and representative selection of samples, leading to improved generalization and reduced bias.[23] In total, 8401 images were categorized and sorted into two distinct datasets, one containing the image deemed "measurable" and the other containing those identified as "visible." "Measurable" images displayed the shrimp's rostrum and telson, while "visible" images lacked one of these features, as shown in Fig. 7. A total of 8401 images formed the basis of this study; of these, 3112 were

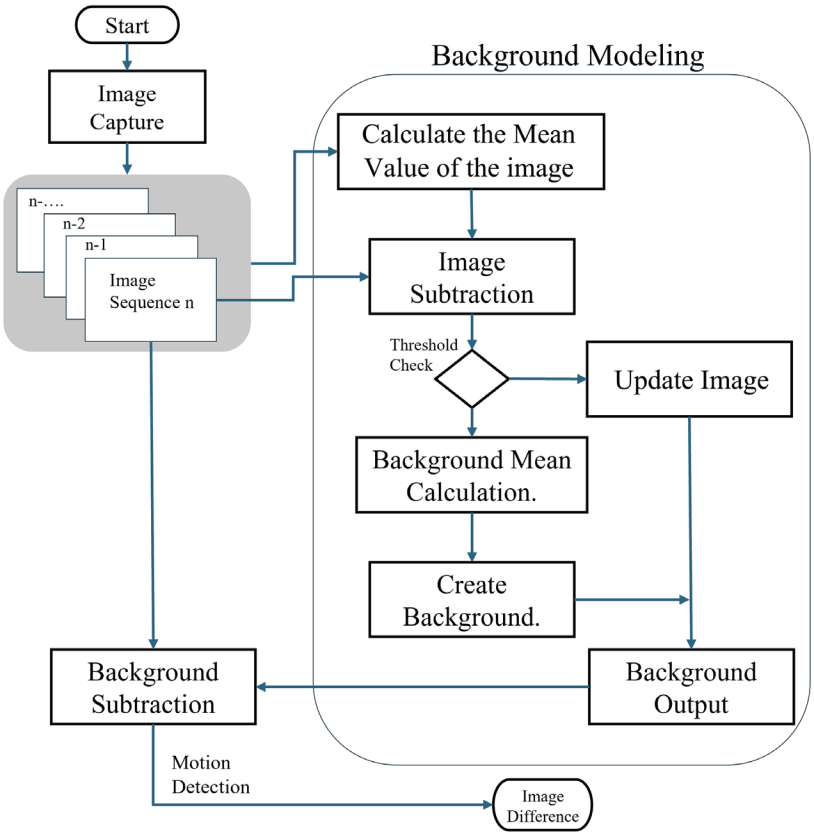


Fig. 6. (Color online) Background modeling process.

Fig. 7. (Color online) Segmented and classified images in the dataset.

suitable for precise measurements, while the remaining 5289 were included in the overall dataset but were not used in the quantitative analysis.

2.3 Data preprocessing and governing

Prior to being fed into the neural network, all images underwent a normalization process, which involved resizing them to a fixed dimension of 128 pixels in width and height, while maintaining three color channels for each image. The initial data processing involved several key steps: first, reading the original image file; second, generating a square-shaped canvas as the foundation for further image manipulation; third, resizing the image to adhere to the predetermined dimensions; fourth, standardizing the pixel values across the entire image to ensure uniformity; and finally, randomly dividing the dataset into distinct training and validation sets for robust model evaluation. The following detailed explanation outlines the procedure to be followed, ensuring clarity and precision in execution.

(1) First, the original image file should be read. The program will then compare the first two dimensions of the image to determine the largest value; this value will be used to create a black canvas of the calculated maximum size.
(2) The process involves generating a square image by overlaying the original image onto a black square canvas, effectively creating a square-shaped composite image.
(3) A square image is required; therefore, the matrix needs to be downscaled to the dimensions of 128 pixels in width, 128 pixels in height, and three color channels.
(4) To promote faster and more effective model convergence in later training stages, we recommend a preprocessing step where each pixel in the image is divided by 255, resulting in standardized pixel values across the entire image.

(5) The data will be randomly divided into training and validation sets employing a secure packaging process to ensure data integrity throughout the entire procedure.

2.3.1 Model structure

To achieve precise shrimp length measurement, the designed CNN architecture incorporated four convolutional layers and two fully connected layers; all layers used ReLU activation functions, a configuration optimized for effective feature extraction. To mitigate overfitting and bolster the robustness of the model, dropout layers were incorporated at a rate of 25%. Following dropout_3, a global pooling layer was substituted for the max pooling layer; this substitution improved feature-category relationships and avoided the necessity of any parameter optimization. The architecture of the recognition model is illustrated in Fig. 8.

2.3.2 Model training

Eighty percent of the complete dataset was allocated to the training phase, while the remaining twenty percent was reserved for validation purposes; this process was carried out using a batch size of 300 instances and a learning rate parameter set at 0.01. For this model, the Adam optimizer was selected to adjust the weights, and the cross-entropy loss function was used to measure the model's performance. To prevent overfitting and ensure optimal performance, the

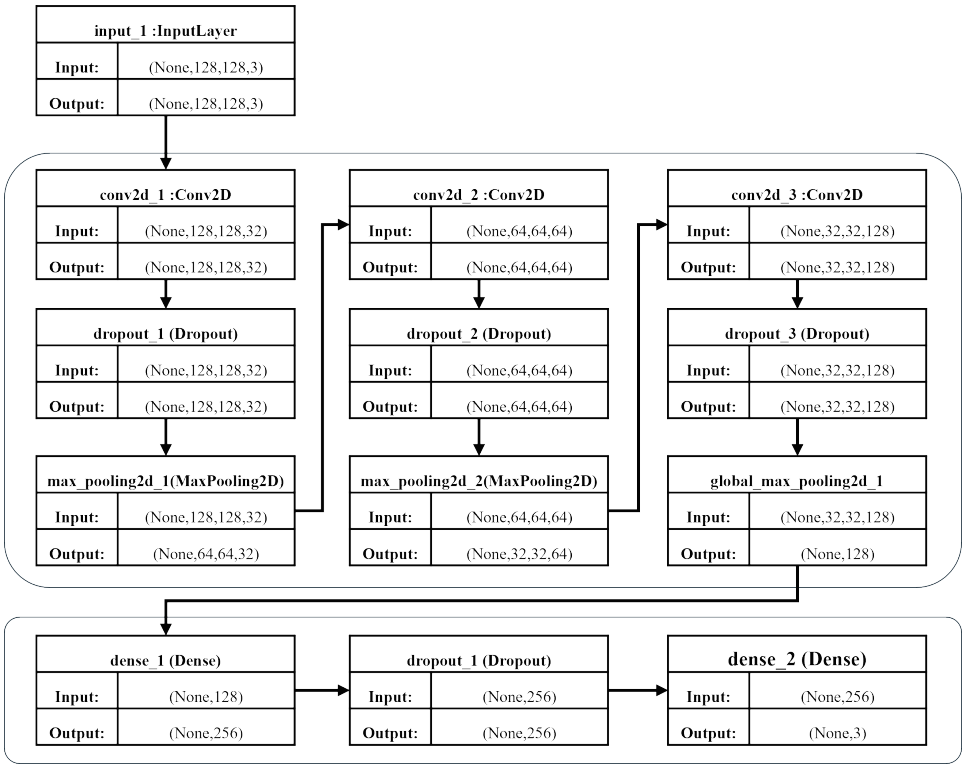


Fig. 8. Recognition model.

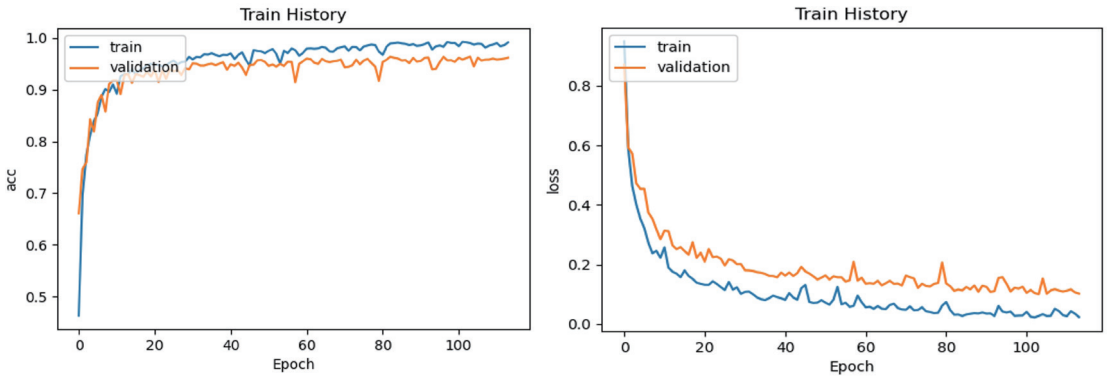
model training was restricted to a maximum of 1000 epochs, incorporating an early stopping mechanism that continuously monitored the validation loss, thereby selecting the model weights that yielded the best performance on unseen data. The results of the learning and validation processes, including accuracy and loss, are displayed and further elaborated in Fig. 9.

2.4 Estimation and analysis

An image-based methodology was employed to collect biological data from the white shrimp, encompassing measurements of length, weight, and other relevant parameters. The acquisition of white shrimp within the market setting implies that the shrimp's size is satisfactory and meets the standards of acceptability for consumption among consumers. From the marketplace, 30 shrimp samples were acquired, with each sample undergoing careful measurement and weighing under controlled lighting conditions to guarantee the precision and accuracy of the data gathered. From the rostrum to the telson, we measured the length; head width was determined by measuring at the heart, and the midpoint served as the measurement point for body width. A ruler, as illustrated in Fig. 10, was included in the image records to show scale, enabling viewers to accurately assess the size and proportions of the objects or features presented.

The collected data were analyzed using multiple regression with length as the independent variable and weight as the dependent variable, establishing a length–weight estimation formula. The relationship is depicted in Fig. 11 with the following equation:

$$y = 0.002x^4 - 0.0578x^3 + 0.7526x^2 - 3.5356x + 5.8716\,. \tag{5}$$



(a) (b)

Fig. 9. (Color online) (a) Accuracy rates and (b) loss rates of learning and validation.

Fig. 10. (Color online) Shrimp size measurement.

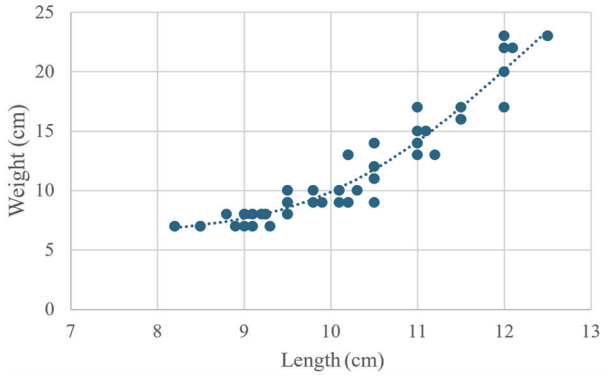


Fig. 11. (Color online) Correlation between length and weight.

3. Results

To avoid disrupting regular aquaculture activities, we conducted tests outdoors on the basis of recommendations from experienced aquaculture personnel and relevant literature. Owing to practical constraints in the aquaculture environment, direct on-site testing could not be conducted to avoid potential bacterial and viral contamination within the shrimp pond. Instead, water samples were collected from the pond by authorized personnel and transported to a controlled laboratory environment. These water samples, including the natural pond color and turbidity, were used to create realistic experimental backgrounds. The setup simulated the actual aquaculture conditions, allowing us to conduct image capture and testing while maintaining a representative environment for shrimp measurement.

3.1 Experimental setup for model training

To avoid negatively impacting the white shrimp population in actual farming situations and to mitigate the resulting financial losses, all mandatory training exercises and photographic documentation will be strictly limited to outdoor tank environments. The experimental design incorporated a series of outdoor tanks, each featuring a water circulation system, and containing varied molasses concentrations to simulate a range of turbidity levels. To create a realistic aquaculture environment for the experiment, we established controlled lighting conditions that mimicked those found in real-world aquaculture facilities. This method ensured the reliability of our experimental results, fully adhering to and meeting all biosafety protocols and standards mandated for the aquaculture site. To effectively simulate and represent the various levels of turbidity, different molasses concentrations were utilized in the experiment. For the test, a 200 L barrel, measuring 81 cm in length, 60 cm in width, and 60 cm in height, was employed, with approximately 100 L of tap water used to fill it, thus establishing the testing environment. To increase oxygen levels, we installed a comprehensive system including underwater filtration equipment, a flow generator, and a high-output air pump capable of delivering up to 300 L of air per hour. As illustrated in Fig. 12, a quantity of 3 g of powdered nitrifying bacteria was added over a seven-day period to facilitate the reduction of ammonia nitrogen concentrations.

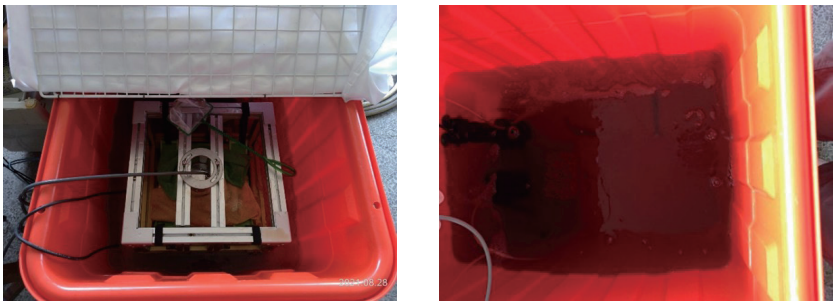


Fig. 12. (Color online) Layout for experiments.

After the initial procedures were completed, the shrimp were added to the water, and then their adaptation was closely monitored and recorded, with the specific indicators used being those outlined in Table 2. The observation platform's data acquisition was successful across a broad spectrum of conditions, including varied background scenarios and the application of numerous watercolor effects, resulting in a total data collection time of 50 h. The negative impact of increased turbidity on image clarity was effectively counteracted through the strategic implementation of black backgrounds in the image acquisition process.

3.2 Effects of background color on average background modeling

To change the background color, we installed physical bottom boards of white, orange, and black in the observation tank. We used a fixed lighting system and an underwater camera together to capture images under stable conditions. This method ensured a stable background, allowing for valid comparisons. We omitted dynamic image processing for the sake of accuracy and simplicity. Average background modeling was used to detect errors in a 5 min video with different background colors and molasses concentrations, simulating aquaculture conditions. Errors were defined as frames with no relevant shrimp images. Figure 13 shows the comparison results and causes.

3.2.1 White background and molasses concentration experiment

The use of a white background in the image resulted in incomplete shrimp segmentation and the detection of shrimp shadows because of light reflection (see Fig. 14 for details). Because a higher concentration of molasses yielded a yellow background, this changed the shrimp's appearance to yellow and therefore made the white background unsuitable for observing the shrimp.

3.2.2 Orange background and molasses concentration experiment

With a deep orange background (Fig. 15) utilized to mitigate the negative effects of light reflections and shadows, we observed a notable improvement in the accuracy of shrimp

Table 2
Shrimp mobility indicators.

Water Situation	No Interference		Human Action		Reaction to Feeding
	Lying on Side	Swimming	Touching	Light and Shadow	
Good			Swim away	Avoid	Look for food close by
Poor			Sluggish	No avoidance	No reaction

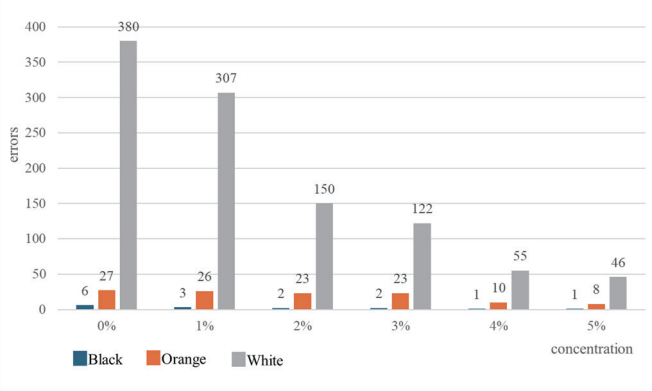


Fig. 13. (Color online) Background color and molasses concentration.

(a) (b) (c)

Fig. 14. (Color online) White background results of segmentation and detection: (a) white background output, (b) binarization, and (c) result.

(a) (b) (c)

Fig. 15. (Color online) Orange background results of segmentation and detection: (a) orange background output, (b) binarization, and (c) result.

detection. With a rise in molasses concentration, the background in the images became increasingly blurry, which in turn led to a decrease in shadow-related issues and a subsequent reduction in the error rates observed during object detection processes.

3.2.3 Black background and molasses concentration experiment

The striking contrast provided by the black background in Fig. 16 considerably minimized shadow interference, resulting in the shrimp being clearly visible and thus allowing for the most complete and accurate image segmentation possible. Despite the pale-yellow hue that the shrimp developed in response to the higher molasses concentration, the stark black background continued to optimize visual contrast, thus ensuring easy observation of the shrimp. The black background offered superior contrast for shrimp image segmentation; however, the substantial light absorption inherent to this background under low-light conditions required additional steps to improve the illumination.

3.3 Model recognition results

The trained model underwent a thorough evaluation process, which involved the use of a validation dataset containing a total of 798 images, to assess its performance. In terms of its overall performance, the model exhibited an average accuracy that reached 95%. The results for precision, recall, and accuracy are summarized and thoroughly detailed in Table 3. The performance of the model was negatively impacted by the overlapping features between the "visible" and "measurable" categories, leading to the "visible" category showing better recall as the model struggled to differentiate between them.

The accuracy of length and weight estimates produced via image processing techniques was evaluated by a direct comparison with the corresponding values obtained through manual data recording. The measurements of length were found to have an error rate that reached 8.4%. Two factors contributed to the 22.0% weight estimation error rate: the regression formula's dependence on length, and the lack of a sufficient quantity of biological sample data, which would have made the model more robust. The growth of shrimp consists of an initial lengthening phase followed by a thickening phase, causing a subsequent increase in the frequency of errors.

(a) (b) (c)

Fig. 16. (Color online) Black background results of segmentation and detection: (a) black background output, (b) binarization, and (c) result.

Table 3
Model confusion matrix.

		Detection Results		
		Measurable	Visible	Precision
Prediction	Measurable	TP	FP	$PV = \frac{TP}{TP + FP}$
		371	10	97%
	Visible	FN	TN	$PV = \frac{TN}{FN + FN}$
		29	388	93%
	Recall	$TPR = \frac{TP}{TP + FN}$	$TPR = \frac{TN}{FP + TN}$	
		92%	97%	
	Accuracy	$ACC = \frac{TP + FN}{T}$	95%	

Through a carefully designed experimental setup and employing innovative methods, we successfully demonstrated the practical viability of leveraging advanced image processing and machine learning techniques to achieve highly accurate measurements of shrimp length and weight. The use of a black background, which proved to be highly effective for observation purposes, resulted in the developed model achieving a high level of accuracy in both the detection and the precise measurement of shrimp.

4. Discussion

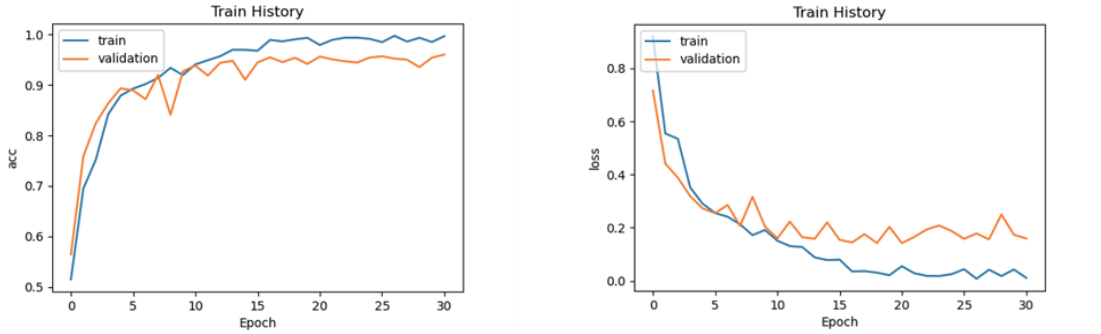
In response to the model structure suggested in Sect. 2.3.1, we made a series of parameter adjustments, which included modifications such as increasing and decreasing the size of the feature extraction layer, altering the dropout layer's parameters, and expanding the capacity of the multilayer perceptron layer to improve performance and accuracy. We implemented the parameter change schemes given in Table 4 while carefully maintaining the original input data dimensions and employing the same training methods throughout the process.

For the parameter combinations in Table 4, the performance of the trained model is shown in Fig. 17, where Fig. 17(a) shows the accuracy curves of the training set and the validation set. An accuracy of 90% on the validation set is insufficient for a complete evaluation of the model's performance capabilities. As can be seen from the loss curve in Fig. 17(b), the loss value of the training set continues to decrease, while the loss value of the validation set tends to be constant, indicating that there is a potential overfitting problem with the model.

A CNN's ability to successfully extract and represent relevant features is significantly impacted by model depth, thus directly affecting the model's overall proficiency and performance. The depth of a model, especially in CNNs, is crucial in feature extraction; more layers allow for the extraction of progressively more intricate and detailed features, building a hierarchical representation of the input data. The process of layering convolutional layers on top

Table 4
Cases of parameters used in the experiment.

	Model from Sect. 2.2.1	Model of modification
Case 1	conv2d_5 (Conv2D) conv2d_6 (Conv2D) global_max_pooling2d	Remove the feature extraction layer and add a flattened layer to connect the multilayer perception layer of the next layer
Case 2	dropout_3(Dropout) (None, 128)	Dropout and perception layers are added to the structure as follows: dense_2 (Dense) (None, 64) dropout_4 (Dropout) (None, 64) dense_3 (Dense) (None, 32) dropout_5 (Dropout) (None, 32)



(a) (b)

Fig. 17. (Color online) (a) Accuracy and (b) loss rates of case 1 with modification model.

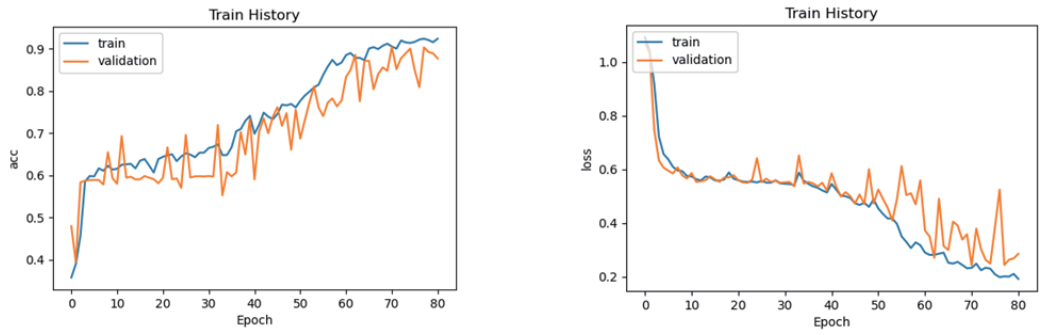
of each other allows for the capture of feature transitions, thereby improving the overall representation of the data.

Increasing the number of layers in a model, however, substantially increases the complexity of the feature space, thereby improving the model's capacity to learn from training data. However, for datasets lacking sufficient complexity, excessively deep models are prone to overfitting, hindering their generalizability to validation data and real-world applications. This is because when the characteristics of the data are not intricate, the model has a tendency to overemphasize the random noise present within the training dataset, thereby hindering its ability to generalize effectively to unseen data and instead focusing on details that are not representative of the broader patterns within the data.

As illustrated by the loss curve in Fig. 17, the training set loss shows a continuous downward trend, whereas the validation set loss plateaus, signifying that the model is increasingly overfitting the training data during the model's learning process. The problem of overfitting in machine learning models can be addressed through the application of regularization techniques; methods such as Dropout and L2 regularization are commonly used to decrease the incidence of overfitting during model training and improve the generalization performance. In addition to its other benefits, the early stopping training strategy includes a mechanism that terminates the training procedure once it is observed that the validation loss is no longer declining, thereby mitigating the risk of model overfitting.

Notably, as the model's architecture expands to incorporate four layers, as illustrated in Fig. 18, while the training set's accuracy and loss curves exhibit a degree of upward progression, the concurrent oscillations become increasingly pronounced and readily apparent. A closer look at the validation set shows a considerable instability in the model's learning, evidenced by pronounced variations in accuracy and loss; this suggests a susceptibility to becoming trapped in a local minimum and an inconsistent ability to extract meaningful features during each training epoch. This phenomenon usually occurs when the depth is large, but the amount of data is small, or the feature differences are unclear. Within the field of deep learning, selecting an appropriate optimization algorithm, such as Adam or other adaptive learning rate methods, is critical for enhancing the model training stability and mitigating the risk of the model converging to suboptimal local minima during the learning process. In addition, gradient clipping is also a commonly used method to prevent the explosion or disappearance of gradients during backpropagation, thereby improving the stability of the training process.

The 22% error in weight estimation is likely a result of the limited number of biological variables, such as width and thickness, which were included in the model. Future development will integrate more features into the system and expand the dataset used for training, ultimately leading to a significant improvement in the accuracy of predictions. In the future, our research efforts will concentrate on the development and implementation of advanced correction algorithms designed to address and mitigate the effects of these previously identified factors. Compared with previous studies, our method achieved higher individual detection accuracy and more precise length measurement. Lai *et al.*[3] reported a detection accuracy of about 88% for surface fish behavior monitoring, while our system achieved 95% accuracy for shrimp identification under turbid underwater conditions. In addition, compared with manual tray observations,[2] our method provides automatic length measurement with only 8.4% error. These results demonstrate the superior performance and practical applicability of our approach. Automated length measurement facilitates real-time monitoring capabilities and enables more effective feed management practices within the aquaculture industry.



(a) (b)

Fig. 18. (Color online) (a) Accuracy and (b) loss rates of case 2 with modification model.

5. Conclusions

In the proposed method, we used advanced hardware design and sophisticated image processing techniques. The observation vehicle and feeding platform were designed to facilitate clear and accurate shrimp observation. The use of CNNs and average background modeling enhanced the precision of moving object detection and measurement, ensuring reliable length and weight estimations. This multifaceted approach provides a robust framework for future research and practical applications in shrimp aquaculture.

We successfully measured the length and estimated the weight of white shrimp by using advanced image processing and machine learning techniques. We designed a specialized observation vehicle and feeding platform, utilizing moving object detection methods and CNNs for shrimp recognition.

The test environment effectively simulated real aquaculture conditions, allowing for controlled testing without disrupting farming operations. Various backgrounds and water conditions aided the detection method evaluation. The black background was the best for observing white shrimp as it reduced shadows and improved segmentation. By combining average background modeling with CNNs, foreground and background were effectively differentiated even under varying underwater color conditions. The results of this study indicate highly accurate shrimp detection, with a 95% accuracy rate and a relatively low length estimation error rate of 8.4%. A 22% error in weight estimation clearly indicates a deficiency in the current model, thereby necessitating additional refinement procedures to achieve a more precise and dependable outcome.

By improving feed management and reducing resource waste, the developed system supports sustainable aquaculture practices. Through the integration of underwater sensors, tailored materials such as feeding mechanisms and observation chambers, and advanced image sensing technology, we demonstrated a comprehensive sensing application for noninvasive aquatic species monitoring. Our future research will focus on incorporating IoT-based real-time monitoring, expanding biological parameters such as shrimp body thickness and width to refine the weight estimation model, and enhancing the system's scalability. Additionally, efforts will be made to enlarge the biological dataset and develop correction algorithms to address the current 22% weight estimation error. The aim of these improvements is to achieve more accurate shrimp growth prediction and enable more efficient and sustainable aquaculture management.

Acknowledgments

We would like to express our sincere thanks to the National Science and Technology Council (NSTC) of Taiwan for supporting this research under project number NSTC 113-2622-E-992-009.

References

1 D. G. Harshith, S. Surve, S. N. S. Prasad, B. V. Ganesh, and K. A. Thomas: Proc. 2023 5th Int. Conf. Bio-Engineering for Smart Technologies (BioSMART) (IEEE, Paris, France, 2023) 1–4. https://doi.org/10.1109/BioSMART58455.2023.10162001
2 C. Nontarit, T. Kondo, M. Yamaguchi, W. Khamkaew, J. Woradet, and J. Karnjana: Proc. 2023 20th Int. Conf. Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (IEEE, Nakhon Phanom, Thailand, 2023) 1–5. https://doi.org/10.1109/ECTI-CON58255.2023.10153372
3 P.-C. Lai, H.-Y. Lin, J.-Y. Lin, H.-C. Hsu, Y.-N. Chu, C.-H. Liou, and Y.-F. Kuo: Biosyst. Eng. **221** (2022) 224. https://doi.org/10.1016/j.biosystemseng.2022.07.006
4 F. D. Von Borstel Luna, E. de la Rosa Aguilar, J. S. Naranjo, and J. G. Jagüey: IEEE Trans. Syst. Man Cybern. Syst. **47** (2017) 1575. https://doi.org/10.1109/TSMC.2016.2635649
5 A. Banwari, R. C. Joshi, N. Sengar, and M. K. Dutta: Ecol. Inform. **69** (2022) 101602. https://doi.org/10.1016/j.ecoinf.2022.101602
6 J. H. Christensen, L. V. Mogensen, R. Galeazzi, and J. C. Andersen: Proc. IEEE/OES Autonomous Underwater Vehicle Workshop (AUV) (IEEE, 2018) 1–6.
7 Y. Zheng, H. Lu, J. Wang, W. Zhang, and M. Guizani: IEEE Trans. Circuits Syst. Video Technol. (2024) 3527. https://doi.org/10.1109/TCSVT.2024.3508102
8 Y. Atoum, S. Srivastava, and X. Liu: IEEE Signal Process. Lett. **22** (2015) 1089. https://doi.org/10.1109/LSP.2014.2385794
9 M. Thilagaraj, S. P. Dhanush, K. Ramaraj, C. Mamani, and A. Patil: Proc. 2023 7th Int. Conf. Design Innovation for 3Cs Compute Communicate Control (ICDI3C) (IEEE, Karnataka, India, 2023) 148–152. https://doi.org/10.1109/ICDI3C61568.2023.00039
10 L. Zheng, M. Gong, Z. Li, and M. Yang: Proc. 2024 IEEE 3rd Int. Conf. Electrical Engineering, Big Data and Algorithms (EEBDA) (IEEE, Changchun, China, 2024) 1169–1173. https://doi.org/10.1109/EEBDA60612.2024.10485688
11 W. Wu and J. Lai: IEEE Access **12** (2024) 15236. https://doi.org/10.1109/ACCESS.2024.3357519
12 B. Y. Sandeep and C. Chandraprakash: IEEE Trans. Instrum. Meas. **73** (2024) 7506308. https://doi.org/10.1109/TIM.2024.3436119
13 Y. Zhang and F. Wang: Proc. 2015 5th Int. Conf. Instrumentation and Measurement, Computer, Communication and Control (IMCCC) (IEEE, Qinhuangdao, China, 2015) 196–199. https://doi.org/10.1109/IMCCC.2015.48
14 Y. Zhou and S. Maskell: Proc. 2019 22nd Int. Conf. Information Fusion (FUSION) (IEEE, Ottawa, ON, Canada, 2019) 1–8. https://doi.org/10.23919/FUSION43075.2019.9011271
15 P. Dewan and R. Kumar: Proc. 2017 Int. Conf. Trends in Electronics and Informatics (ICEI) (IEEE, Tirunelveli, India, 2017) 651–656. https://doi.org/10.1109/ICOEI.2017.8300783
16 B. Khagi and G. R. Kwon: IEEE Access **9** (2021) 96930. https://doi.org/10.1109/ACCESS.2021.3093455
17 J. A. Hertz: Introduction to the Theory of Neural Computation (CRC Press, Boca Raton, 2018).
18 A. Mollahosseini, D. Chan, and M. H. Mahoor: Proc. Winter Conf. Applications of Computer Vision (WACV) (IEEE, 2016) 1–10.
19 A. G. Chigozie, E. Nwankpa, W. Ijomah, and S. Marshall: arXiv (2018). https://arxiv.org/abs/1811.03378
20 A. A. Mohammed and V. Umaashankar: Proc. 2018 Int. Conf. Advances in Computing, Communications and Informatics (ICACCI) (IEEE, Bangalore, India, 2018) 1090–1094. https://doi.org/10.1109/ICACCI.2018.8554637
21 S. H. Chan, T. Zickler, and Y. M. Lu: IEEE Trans. Image Process. **26** (2017) 5107. https://doi.org/10.1109/TIP.2017.2731208
22 X. Zhao, S. Yan, and Q. Gao: IEEE Access **7** (2019) 15018. https://doi.org/10.1109/ACCESS.2019.2895072
23 C. Wang, D. Xu, R. Wan, B. He, B. Shi, and L.-Y. Duan: IEEE Trans. Multimedia **25** (2023) 2876. https://doi.org/10.1109/TMM.2022.3152390

About the Authors

Shao-Yong Lu earned his college degree from National Kaohsiung Institute of Technology in 1991, his B.S. degree from Feng Chia University in 1993, and his M.S. degree from National Taiwan Institute of Technology in 1995. Since 1987, he has been working at China Steel Corp. and is now General Manager of China Prosperity Development Co., Ltd. Currently, he is a Ph.D. student at National Kaohsiung University of Science and Technology (NKUST), specializing in intelligent IoT, data analytics, and sustainable energy engineering (i112154102@nkust.edu.tw)

Yu-Sheng Tu received his B.S. degree from National Ilan University in 2019 and his M.S. degree in electrical engineering from National Kaohsiung University of Science and Technology in 2021. Since 2023, he has been a system maintenance engineer at Taiwan Power Company. His research interests include AI, automation, and data analysis, with a focus on enhancing equipment reliability and system maintenance. He hikes, explores technology, and values learning. (parab2455@gmail.com)

Wen-Ping Chen is a professor in the Department of Electrical Engineering at National Kaohsiung University of Science and Technology, Taiwan. His primary research interests include the development of intelligent robots, IoT applications, design and implementation of automated processes, automated optical inspection and its applications, design and development of communication protocols, intelligent disaster prevention and emergency medical notification, and systems integration. (wpc@nkust.edu.tw)